1262651

THE UNITED STATES OF AMERICA

TO ALL TO WHOM THESE PRESENTS SHALL COME:

UNITED STATES DEPARTMENT OF COMMERCE

United States Patent and Trademark Office

December 17, 2004

THIS IS TO CERTIFY THAT ANNEXED HERETO IS A TRUE COPY FROM THE RECORDS OF THE UNITED STATES PATENT AND TRADEMARK OFFICE OF THOSE PAPERS OF THE BELOW IDENTIFIED PATENT APPLICATION THAT MET THE REQUIREMENTS TO BE GRANTED A FILING DATE.

APPLICATION NUMBER: ***60/504,380***
FILING DATE: ***September 17, 2003***
RELATED PCT APPLICATION NUMBER: ***PCT/US04/30580***

Certified by

Jon W Dudas

Acting Under Secretary of Commerce
for Intellectual Property
and Acting Director of the U.S.
Patent and Trademark Office

UNITED STATES PATENT AND TRADEMARK OFFICE
DEPARTMENT OF COMMERCE

Knobbe Martens Olson & Bear LLP

Intellectual Property Law

2040 Main Street
Fourteenth Floor
Irvine, CA 92614
Tel 949-760-0404
Fax 949-760-9502
www.kmob.com

Lee W. Henderson, Ph.D.

MAIL STOP PROVISIONAL PATENT APPLICATION
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

CERTIFICATE OF MAILING BY "EXPRESS MAIL"

Attorney Docket No.	:	UCLARF.003PR
Applicants	:	Verbauwhede et al.
For	:	A DYNAMIC AND DIFFERENTIAL CMOS LOGIC WITH SIGNAL INDEPENDENT POWER CONSUMPTION TO WITHSTAND DIFFERENTIAL POWER ANALYSIS
Attorney	:	Lee W. Henderson Ph.D.
"Express Mail" Mailing Label No.	:	EV 320128401 US
Date of Deposit	:	September 17, 2003

I hereby certify that the accompanying

Transmittal letter; specification (including Appendix A) and drawings in 56 pages; Check for Filing Fee; Return Prepaid Postcard

are being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and are addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

Nelson Merida

H:\DOCS\LWH\LWH-9208.DOC
091703

PROVISIONAL APPLICATION FOR PATENT
COVER SHEET

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

ATTENTION: PROVISIONAL PATENT APPLICATION

Sir:

This is a request for filing a PROVISIONAL APPLICATION FOR PATENT under 37 CFR § 1.53(c).

For: A DYNAMIC AND DIFFERENTIAL CMOS LOGIC WITH SIGNAL INDEPENDENT POWER CONSUMPTION TO WITHSTAND DIFFERENTIAL POWER ANALYSIS

Name of First Inventor: Ingrid Verbauwhede

Name of Second Inventor: Kris Tiri

Enclosed are:

(X) Specification (including Appendix A) and drawings in 56 pages.

(X) The present application qualifies for small entity status under 37 CFR 1.27.

(X) A check in the amount of $80 to cover the filing fee is enclosed.

(X) A return prepaid postcard.

(X) The Commissioner is hereby authorized to charge any additional fees which may be required, now or in the future, or credit any overpayment to Account No. 11-1410.

Was this invention made by an agency of the United States Government or under a contract with an agency of the United States Government?

( ) No.

(X) Yes. The name of the U.S. Government agency and the Government contract number are: NSF, Contract No. CCR-0098361

(X) Please send correspondence to:

Lee W. Henderson Ph.D.
Knobbe, Martens, Olson & Bear, LLP
2040 Main Street, 14th Floor
Irvine, CA 92614

Respectfully submitted,

Lee W Henderson

Lee W. Henderson Ph.D.
Registration No. 41,830
Customer No. 20,995
(949) 760-0404

H:\DOCS\LWH\LWH-9207.DOC
091703

A DYNAMIC AND DIFFERENTIAL CMOS LOGIC WITH SIGNAL INDEPENDENT POWER CONSUMPTION TO WITHSTAND DIFFERENTIAL POWER ANALYSIS

Summary

1. Introduction

Electronic banking, e-commerce, virtual private networks and so on cannot operate without encryption technology and a secure implementation of the encryption technology. To obtain security, many strong encryption algorithms have been developed. While usually strong against mathematical attacks, side channel attacks can reveal the secret key through information leaked by the hardware implementation of the encryption module. Differential Power Analysis (DPA) is based on the fact that logic operations have power characteristics that depend on the input data: statistical analyses of measured power traces link the switching activities of the circuit to the secret key.

Different techniques have been tried to prevent this information leakage. On the algorithmic level, random process interrupts interleave dummy instructions to avoid sequential execution of the algorithm. Integration techniques, however, are able to resynchronize the power traces. Masking is a technique that prevents intermediate variables from depending on the knowledge of an easily-accessible subset of the secret key. DPA has been modified to handle masking. On the architectural level, techniques include adding random power consuming operations and duplicating logic with complementary operations. These procedures merely lower the side channel information and might easily be disabled through tampering. Active power signal filtering with power consumption compensation, passive filtering, battery on chip and detachable power supply influence the power transfer itself. The first method lags behind the fast power fluctuations and physical dimensions limit the latter three.

The foregoing methods attempt to conceal the supply current variations at the architectural or the algorithmic level. However the variations originate at the logic level. One way to halt DPA is to have the encryption module or at least the sensitive parts of it implemented in a logic, whose power consumption is independent of the signal transitions. As described in section 2 below, differential logic, which is often disclosed as a solution, does not possess such power characteristics. The following section develops a dynamic and differential CMOS logic style starting from the problem setting of the DPA. Next, the disclosed logic style is compared with Static Complementary CMOS for power variation, power consumption and area. Finally a conclusion is formulated.

2. Imperfections in existing logic and derivation of the dynamic and differential logic

As shown in Figure 1, Static Complementary CMOS logic (scCMOS), which is the default logic style in standard cell libraries used for security IC's only consumes energy from the power supply when its output has a 0-1 transition. During the 1-0 transition, the energy previously stored in the output capacitance is dissipated. In the two degenerated events of 0-0 or a 1-1 transition no power is used. This asymmetric power profile provides the information used in DPA to find the secret key.

The logic style disclosed herein with data-independent power consumption does not reveal this information. In one embodiment logic values are measured by charging and discharging capacitors using a fixed amount of energy for every transition. In one embodiment, even though different capacitances are switched, the logic style provides the property of charging in every cycle a total capacitance with a constant value. The disclosed Sense Amplifier Based Logic (SABL) achieves this goal by (1) switching the output independently of the input value, sequence and by (2) having a relatively constant load capacitance equal to all internal nodes combined with one of the balanced output loads.

In SABL, the combination of dynamic and differential logic charges a capacitance for the four output transitions (0-0, 0-1, 1-0, 1-1). Figure 2 shows the output events for differential logic and for dynamic logic separately. A differential logic masks the input value: independent of the input value, energy is dissipated when exactly one output node is discharged. Therefore there is no difference between 0-1 and a 1-0 event or between a 0-0 and a 1-1 event. However one can still differentiate between those two main classes: a 0-1/1-0 transition will consume power whereas a 0-0/1-1 not. Note that this is the main reason that addressing the power attack solely by balancing the Hamming weights does not succeed. Whether it is done on algorithmic level (e.g., exclusively handling bytes with Hamming weight 4), architecture level e.(e.g., duplicating the module with a complementary module) or logic level (e.g., differential logic), this difference will remain present.

A dynamic logic breaks the input sequence: independent of the input-switching behavior, energy is consumed when the load capacitance is charged. Therefore there is no difference between a 0-0 and a 1-0 event or between a 1-1 and a 0-1 event. Here, only the 0-1 and 1-1 transition will induce power consumption during the following precharge phase. Thus, it is useful to combine the two into one dynamic differential logic style that switches the output independently of the input value or sequence.

Merely making it dynamic and differential is not sufficient however, as it makes the four transitions equal but only to the first order. This will be shown for the dynamic DCVSL style. The DCVSL inverter has a very regular power consumption: simulations show a variation smaller than 1%. But for more complicated logic functions this number will not be accurate. Figure 3 shows the DCVSL AND-NAND gate, for which simulations indicate that the difference can be as large as 50%. This is caused by asymmetry in the gate. Depending on the input, different parasitic capacitances discharge during the evaluation phase. In the succeeding power consuming precharge phase, these capacitances are recharged. In none of the four different events, the same combination of capacitances has to be charged.

SABL makes the four output events equal, by charging at every event the same capacitance value: one of the balanced load capacitances and the sum of all the internal node capacitances.

2.1. SABL: basic gate .

The SABL gate is based on the StrongArm110 flipflop (SAFF). To realize a basic gate, keep the sense amplifier half of the flip-flop and replace the input differential pair by a differential pull down network (DPDN). Figure 4 depicts the generic n-gate. The DPDN is implemented such that for a stable input combination all nodes that are internal to the DPDN connect to one of the output nodes. During evaluation (clk high) the crosscoupled inverter will toggle to one state and provide a stable output as soon as the DPDN provides a path to ground.

Transistor M_1, which is always on, prevents a floating node by serving as a path for subthreshold currents, as it does in case of the original SAFF. In addition, in case of the SABL gate, M_1 guarantees that all internal nodes discharge. Regardless of which branch is on, all internal nodes and their respective capacitances are connected through M_1 and will eventually be discharged together with one of the output nodes. The differential output nodes connect differential signals to a differential input. Under the assumption that the differential signals travel in the same environment, the interconnect capacitances are equivalent. Therefore the total output capacitance, including the equal intrinsic output capacitances, one of the interconnect capacitances and one of the symmetrical input capacitances, is a constant. During precharge (clk low) all the discharged nodes and capacitances will be precharged. As such every cycle the same capacitances are discharged and charged what makes the power consumption of the gate independent of the input statistics.

A generic p-gate is implemented as a gate that precharges to GND when clk is high and evaluates one node to VDD through a DPUN when clk is low. Figure 4 illustrates the implementation of an AND-NAND n-gate. Figure 5 shows the discharging and charging events of the AND-NAND n-gate for different inputs.

2.2. SABL: cascading gates

As it is a dynamic logic, SABL is connected using either Domino or np-logic. In case of Domino logic, the use of static inverters between gates does no harm: in every cycle exactly one inverter will have a 0-1 event.

2.3. SABL: storage

The Set-Reset latch of the SAFF is static to prevent loss of the output value. However if the input to the flip-flop does not change, the output of the latch will not change. Consequently there will be no power consumption in this static gate.

A combination of a p-SA and a n-SA, as shown in Figure 6, acts as a master-slave flip-flop. The p-SA evaluates at the falling edge of clk and keeps this value till the rising edge, while the n-SA evaluates at the rising edge of clk and keeps this value till the falling edge. As a result, the value is stored during one clock cycle. For correct operation, the actual precharge time has to be large enough to evaluate the correct differential input. If necessary, this delay can be implemented with static inverters. Note that each SA can as well be replaced with a module of cascaded gates.

3. Experimental results

Encryption algorithms typically use arithmetic that is different from the twos complement arithmetic on integers or real numbers. Instead they use Galois field arithmetic and operations such as substitutions and permutations. The operations can be implemented with the following set of basic cells: inverter, NAND, XOR and flip-flop (FF). In one embodiment, a substitution box (S9-box) has been implemented. The S9-box is a main component of the Kasumi algorithm, the encryption algorithm in 3G cellular standard. The block includes 5 inverter's, 86 NAND's and 92 XOR's and replaces 9 input bits with 9 altered bits.

In one embodiment, circuits are designed in a 0.18-μ.m, 1.8V CMOS technology. In one sample embodiment, a Layout is created with LayoutPlus from CADENCE software. Figure 7 shows the layouts of the S9-box. Layout to netlist is done with Analog Artist of CADENCE. Simulations were done in Hspice, using the LEVEL 49 transistor model. Basic cells were simulated with a fan out of 4 inverters. The variation on the power consumption was measured for a random input sequence of length 300 and 500 clock cycles for basic cells and S9-box respectively.

The power consumption is represented by the energy per cycle and as a measure for the variation on the power consumption one can define the normalized energy deviation:

$$NED = \frac{Max(energy/cycle) - min(energy/cycle)}{Max(energy/cycle)} \qquad (1)$$

NED ranges from 0 to 1. The smaller the variation, or in other words NED, the more measurements are necessary and the more accurate the measuring devices have to be in order to extract the side-channel information. In case of the S9-box, where an event is the sum of a relatively large number of independent identical distributions, the distribution will tend towards the normal distribution Here the normalized standard deviation (NSD), the standard deviation divided by the mean, is also given.

Figure 10 shows that SABL behaves as expected. NED is reduced from more than 80% for scCMOS to below 3% for SABL. NSD decreases from 29% to 0.6%. For the S9-box, SABL uses less than two times the area and the energy of scCMOS. In comparison, the ineffective technique of doubling the logic with complementary logic requires twice the area and energy. Figure 8 shows a histogram of the observed energies per cycle for the S9-box. Figure 9 is a superposition of the power supply current for successive cycles of the simulated transient response, showing that while the observed energies are spread out in a broad range for scCMOS, they remain in a narrow band for SABL. SABL is subject to only minor delay variations as the same amount of charge goes through similar paths during both precharge and evaluation. Furthermore, the histogram shows that the observed energies of SABL are situated near the maximum energy of scCMOS. In comparison, the active power filtering technique will at least require this maximum energy.

Scheduling operations, which have different power characteristics at different instances of time, do not influence the power consumption of a SABL design. Indeed, every gate evaluates at the clock edge, whether or not actual data is processed. Furthermore, clock gating can be applied if the operations scheduled at a certain instance of time are independent of the internal data. This means that in the controller finite state machine, conditional if-then-else branches can be masked by activating the combination of submodules used in each branch and by adding idle states if one branch is longer than the other.

4. Conclusions

A dynamic and differential CMOS logic style is disclosed in which a gate uses a fixed amount of energy per evaluation event. To this purpose, the gate switches its output at every event and loads at that instant a constant capacitance. Experimental results demonstrate a normalized energy variation that is up to 116 times less pronounced when compared to scCMOS implementations while using only two times the area and power. The logic style is a Dynamic and Differential Logic (DDL) style. Independently of the input signals, DDL style logic has one charging event per clock cycle. The differential feature masks the input value since exactly one of the precharged output nodes is discharged during the evaluation phase. The dynamic feature breaks the input sequence: the discharged node is charged during the subsequent precharge phase.

A differential pull down network (DPDN) is used such that every charging event the same capacitance value is charged. The capacitance value includes one of the balanced output capacitances and the sum of the internal node capacitances. The capacitances at the differential output nodes are balanced as all its components, which are the intrinsic output capacitances, the interconnect capacitances and the input capacitances, can be balanced, assuming a careful layout. The DPDN is designed such that, for a stable input combination during the evaluation phase, all nodes that are internal to the DPDN connect to one of the output nodes of the DPDN. As a result, since both output nodes of the DPDN eventually discharge, all the internal nodes are discharged and will be charged during the subsequent precharge phase.

The logic style is fully elaborated: logic gates, combinatorial logic and sequential logic can be implemented. Combinatorial logic can be built using either Domino or np-logic. Sequential logic can be implemented with additional dummy circuitry that switches when the actual flip-flop doesn't switch or with a master-slave flip-flop that stores the value during the precharge phase.

Detailed Description

Outline

1. SABL n-gate
2. Special differential pull down network
    - 2.1. Special DPDN
    - 2.2. Design of special DPDN
    - 2.3. Enhanced special DPDN
3. SABL p-gate
4. Combinatorial logic: cascading gates
5. Sequential logic for storage
    - 5.1. SAFF with additional dummy circuitry
    - 5.2. Master-slave flip-flop
    - 5.3. Design rules: cascading flip-flop's and combinatorial logic
6. Charge Recycling SABL

1. SABL n-gate

In one embodiment, the SABL n-gate is based on the StrongArm110 Flip-Flop (SAFF). Figure 11 depicts the generic n-gate. To realize a n-gate, we keep the Sense Amplifier half (SA) of the SAFF and replace the input differential pair by a differential pull down network (DPDN). The DPDN has two branches, which for a stable differential input combination connect one of the output nodes X and Y of the DPDN to the common node Z.

Fundamentally, the operation of a SABL n-gate is similar to the operation of the SA in the SAFF. When the clk-signal becomes high, which is called the evaluation phase, and as soon as a single branch of the DPDN becomes active such that it provides a path to GND, the cross-coupled inverter will toggle to one state and provide a stable output. In case of the SAFF, transistor M_1, which is always on, prevents a floating node by serving as a path to ground for leakage currents when the inputs to the differential pair would switch after the SA has evaluated. In case of the SABL n-gate, the inputs to the DPDN do not change after the n-gate has evaluated since they come from a stable output of a previous SABL gate. In the present design, M_1 guarantees that both external nodes X and Y of the DPDN discharge. Regardless of which branch of the DPDN is on, X and Y are connected through M_1, and will eventually be discharged together with one of the output nodes. During the next phase, as the clk-signal is low and which is called the precharge phase, all the discharged nodes and capacitances will be charged.

The SABL gate can be used for DPA-resistant logic. Traditionally SA is used in memory designs or high-speed logic.

2. Special differential pull down network

Note that the capacitances at the differential output nodes are balanced as all its components, which will be the intrinsic output capacitances, the interconnect capacitances and the input capacitances, can be balanced by a careful layout. To control the contribution of the parasitic capacitances at the internal nodes of the DPDN, however, a special DPDN is described below.

2.1. Special DPDN

The special DPDN is designed such that for a stable input combination during evaluation phase nodes that are internal to the DPDN connect to one of the output nodes of the DPDN. As a result, since both output nodes of the DPDN eventually discharge, the internal nodes are discharged and will be charged during the subsequent precharge phase. That means that SABL charges every cycle the same capacitance value: one of the balanced load capacitances and the sum of all the internal node capacitances. And hence SABL has signal-independent power consumption.

Figure 12 shows the transformation of a DPDN to a special DPDN used in SABL. The internal node at location n of the DPDN used in And-Nand gates, can be connected to output y without changing the functionality. Figure 13 shows different SABL gates: And-gate, Xor-gate and Or-gate. Figure 14 shows the discharging and charging events of the AND-NAND n-gate for different inputs . The simulations indicated that every cycle the same amount of charge is used.

2.2. Design of special DPDN

The design goal is to guarantee that all internal nodes are connected to one of the external nodes for a differential input. The design procedure is a transformation that repositions transistors in the DPDN. As a result, the total number of devices remains the same. The total evaluation depth may increase.

The design procedure of creating a special DPDN for a logical function f consists of 5 steps:

1. Identify 2 expressions x and y that combine to the logical function f. This results in an AND-operation, $x.y$, or an OR-operation, $x+y$.
2. Complement the expression in x and y to get the dual expression of $\overline{f}$. This results in an OR-operation, $\overline{x}+\overline{y}$, or an AND-operation, $\overline{x}.\overline{y}$ respectively.
3. Transform the OR-operation.

   The results of step 1. and step 2. are two dual expressions:

   either case A) $\begin{cases} f = x.y \\ \overline{f} = \overline{x}+\overline{y} \end{cases}$ or case B) $\begin{cases} f = x+y \\ \overline{f} = \overline{x}.\overline{y} \end{cases}$

One expression is an AND-operation, the other an OR-operation. In the DPDN, the AND-operation is implemented as a series combination, the OR-operation as a parallel combination. At this abstraction level, only the series combination has an internal node.

In case A), we transform the parallel connection into $\bar{x}.y + \bar{y}$, put network y at the bottom of the $x.y$ connection and share network y between the two branches $x.y$ and $\bar{x}.y+\bar{y}$.

In case B) we transform the parallel connection into $x.\bar{y} + y$, put network $\bar{y}$ at the bottom of the $\bar{x}.\bar{y}$ connection and share network $\bar{y}$ between the two branches $\bar{x}.\bar{y}$ and $x.\bar{y}+y$.

Now the DPDN connects the internal node of the series connection to the output node.

4. Repeat the procedure for the logical expressions x and y till the network consists of only 1 transistor.
5. Substitute the results.

For a given schematic of a DPDN, the design procedure translates to (1) identify all networks in series; (2a) open the corresponding parallel connections at the bottom of the network and (2b) connect them to the internal nodes of the series connections; and (3) unroll the network.

Figure 15 illustrates the design procedure applied to a complex DPDN. In the final DPDN, both the true and the inverse of a signal control a transistor for every internal node: independent of the input, every internal node is connected to another node, which is either an external node or another internal node, for which both the true and the inverse of a signal control a transistor. As a result independent of the input, every internal node is connected to an external node.

The transistors in the DPDN can only charge the internal nodes as long as they are on. To assure that every cycle the same amount of charge is consumed, this charge time must be constant. That is the case if at every node both the signal and the inverse of the signal control a transistor that loads the node. Whether it is the transistor controlled by the signal or the one controlled by the inverse, the total charge time for the internal nodes is the time needed to precharge the outputs of the preceding gate. This requirement is fulfilled by the special DPDN.

2.3. Enhanced special DPDN

The SABL logic gate has constant power consumption. Enhancements to the special DPDN, however, are still possible. In the enhanced special DPDN, dummy transistors are inserted in the DPDN. The dummy transistors form a so-called pass-gate, which is a connection between two nodes that is always open for a stable differential input combination. In one embodiment, the dummy transistors are inserted if different discharge paths do not have the same number of transistors, as e.g. is the case with the SABL And-Nand gate. Advantages are that no evaluation will start before all inputs are stable and that for every possible discharge event, there is a constant resistance in the discharge path. Figure 16 shows the SABL And-Nand gate with enhanced special DPDN.

3. SABL p-gate

Figure 12 shows the SABL, p-gate. This gate precharges to GND when the clk-signal is high and evaluates one node to VDD through the differential pull up network (DPUN) when clk-signal is low. Fundamentally, the operation is the same as that of the generic n-gate.

Domino logic and np logic are known construction rules to connect dynamic logic gates together. These rules remain applicable to the SABL gates.

4. Combinatorial logic: cascading gates

As it is a dynamic logic, SABL is typically connected using either Domino logic or np-logic. Both are depicted in Figure 18. In case of Domino logic, the use of static inverters between gates does no harm: in every cycle, one inverter will have a 0-1 event.

5. Sequential logic for storage

The original SAFF uses a static Set-Reset latch to hold the output value during precharge phase of the SA. However if the input to the flip-flop does not change, the output of the latch will not change. Consequently there will be no power consumption in this static gate and hence the gate is vulnerable to a DPA attack. There are 2 solutions to this: (1) an additional dummy latch that switches when the actual latch doesn't switch; or (2) a master-slave flip-flop that stores the value during the precharge phase.

5.1. SAFF with additional dummy circuitry

Dummy circuitry can be introduced that will switch when the latch does not switch and vice versa. Figure 19 depicts a n-flip-flop with dummy circuitry. The idea is to change the input to the dummy latch when the input to the main SR latch does not change. And vice versa, when the input to the main latch changes, the input to the dummy latch remains the same. A n-flip-flop with dummy circuitry stores the input signal at the rising edge of the clk-signal and keeps this value at the output node till the next rising edge of the clk-signal. The dual case is the p-flip-flop with dummy circuitry. This circuit is achieved by changing the n-gates in Figure 19 with p-gates, and the nand/nand static Set Reset latches with nor/nor static Set Reset latches. A p-flip-flop with dummy circuitry stores the input signal at the falling edge of the clk-signal and keeps this value at the output node till the next falling edge of the clk-signal.

5.2. Master-slave flip-flop

A combination of a p-SA and a n-SA, as shown in Figure 20, acts as a master-slave p-flip-flop. The p-SA evaluates at the falling edge of clk and keeps this value till the rising edge, while the n-SA evaluates at the rising edge of clk and keeps this value till the falling edge. As a result, the value is stored during one clock cycle. For correct operation, the actual precharge time has to be large enough to evaluate the correct differential input. If the precharge time is not large enough, a delay Δt is added at the output of the p-SA. This can be done with an extra load capacitance or with static inverters. Note that if the flip-flop is followed by n-gates, both output nodes need to be inverted as has been described in section 4, on cascading gates.

The dual case is the master-slave n-flip-flop. This circuit is achieved by interchanging the n-SA and p-SA in Figure 20. The n-SA evaluates at the rising edge of clk and keeps this value till the falling edge, while the p-SA evaluates at the falling edge of clk and keeps this value till the rising edge. As a result, the value is stored during one clock cycle.

5.3. Cascading flip-flop's and combinatorial logic

SABL logic that evaluates during clk '1' and precharges during clk '0' (e.g. n-gates) is cascaded with p-flip-flop's, which store the value at the falling edge of the clock, as is depicted in Figure 21.

SABL logic that evaluates during clk '0' and precharges during clk '1' (e.g. p-gates) is cascaded with n-flip-flop's, which store the value at the rising edge of the clock.

6. Charge Recycling SABL

In addition to the SABL gate, a Charge Recycling SABL (CRSABL) gate is disclosed herein. The CRSABL n-gate is shown in Figure 22. The gate is based on the SABL gate. Only difference is that the two clocked transistors that precharge the output and the internal nodes are replaced by one clocked transistor between the output nodes. When the clk-signal becomes low, which is called the precharge phase, the charge stored at one output node is recycled to partially charge the output and the internal nodes to an intermediate voltage. During the next phase, as the clk-signal becomes high, which is called the evaluation phase, and as soon as a single branch of the DPDN becomes active such that it provides a path to GND, the cross-coupled inverter toggles to one state and provide a stable output.

The above discussions of sections 2 through 5 are applicable to CRSABL. Attention should be given, however, to see that the intermediate voltages do not falsely evaluate the next gate if np-logic is used and that they do not cause static current consumption if domino logic is used. Solutions are high Vt transistors or using circuits that convert the intermediate voltage to a full rail voltage.

In one embodiment, the SABL gates are used in smart cards and other small embedded devices. In these devices, the power can easily monitored because there are few physical barriers.

The logic style described above not only has a power consumption independent of the input value and sequence but also circuit characteristics, such as leakage current, delay and instantaneous current are independent of inputs and sequences. Therefore, implementing the encryption module in this logic protects it against a wide class of Side Channel Attacks, based on timing, power and leakage information.

Implementing a design in SABL has the additional advantage that the original cryptographic algorithm can be handed over to the hardware engineer without modifications and that subsequently the hardware engineer can make a straightforward logic design. Indeed, whether or not useful data is processed, every SABL gate on the IC evaluates at its particular instance of time. Additional embodiments, features, and aspects are described in Appendix A attached hereto, which forms part of this disclosure.

WHAT IS CLAIMED IS:

1. An apparatus comprising: a sense amplifier based logic gate having an input network, said input network comprising a special differential pull down network configured to connect a plurality of internal nodes of the differential pull down network to an external node such that internal nodes are discharged and will be charged during a subsequent precharge phase to provide signal-independent power consumption.

2. The apparatus of Claim 1, wherein said special differential pull down network comprises an enhanced special differential pull down network that uses dummy transistors to form a pass-gate which is always open.

3. The apparatus of Claim 1, wherein said differential pull down network comprises an enhanced special differential pull down network that uses dummy transistors to form a pass-gate which is always open, said pass gate inserted if different discharge paths have unequal numbers of transistors.

Document made available under the Patent Cooperation Treaty (PCT)

International application number: PCT/US04/030580

International filing date: 17 September 2004 (17.09.2004)

Document type: Certified copy of priority document

Document details: Country/Office: US
Number: 60/504,380
Filing date: 17 September 2003 (17.09.2003)

Date of receipt at the International Bureau: 23 December 2004 (23.12.2004)

Remark: Priority document submitted or transmitted to the International Bureau in compliance with Rule 17.1(a) or (b)

World Intellectual Property Organization (WIPO) - Geneva, Switzerland
Organisation Mondiale de la Propriété Intellectuelle (OMPI) - Genève, Suisse